# KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT yang telah memberikan kesehatan dan rahmat sehingga penulis dapat menyusun dan menyelesaikan laporan tugas besar kelompok praktikum algoritma dan struktur data dan penulis ucapkan terima kasih kepada dosen praktikum algoritma dan struktur data, bapak M.Rahmadhani Iqbal yang telah membimbing penulis dalam penyusunan laporan tugas besar kelompok ini.

Algoritma adalah urutan langkah untuk menyelesaikan masalah secara sistematis dan logis. Algoritma menawarkan suatu metode dalam menyelesaikan sebuah permasalahan. Algoritma diartikan sebagai urutan langkah dalam menyelesaikan masalah secara sistematis dan logis. Pendekatan secara sistematis dan logis tersebut, menjadikan proses penyelesaian masalah terjaga kebenarannya karena algoritma hendaklah benar agar dapat menghasilkan keluaran/solusi yang benar pula.

Akhir kata penulis mengucapkan kepada semua pihak yang telah membantu dalam pembuatan laporan ini, penulis juga menyadari bahwa laporan ini masih jauh dari sempurna dan perlu adanya perbaikan di sana-sini, untuk itu penulis berharap kritik maupun saran dari pembaca yang bersifat membangun. Terima kasih, semoga laporan tugas besar kelompok ini dapat bermanfaat bagi semua pihak khususnya penulis.

Bandung, Desember 2013

Penulis

# DAFTAR ISI

[KATA PENGANTAR i](#_Toc375902425)

[DAFTAR ISI ii](#_Toc375902426)

[BAB I PENDAHULUAN 3](#_Toc375902427)

[A. Latar Belakang 3](#_Toc375902429)

[B. Rumusan Masalah 4](#_Toc375902430)

[C. Manfaat Program 4](#_Toc375902433)

[BAB II LANDASAN TEORI 5](#_Toc375902436)

[A. Sorting 5](#_Toc375902438)

[B. Searching 5](#_Toc375902439)

[C. Linked List 6](#_Toc375902440)

[BAB III PEMBAHASAN PROGRAM 7](#_Toc375902446)

[A. Penjelasan isi dan alur program 7](#_Toc375902448)

[B. Diagram Alir ( Flow Chart ) 9](#_Toc375902456)

[C. Source Code dan Penjelasan Program 10](#_Toc375902457)

[D. Tampilan Hasil Eksekusi 22](#_Toc375902458)

[BAB IV KESIMPULAN DAN SARAN 25](#_Toc375902465)

[A. Kesimpulan 25](#_Toc375902467)

[B. Saran 25](#_Toc375902468)

[DAFTAR PUSTAKA 26](#_Toc375902471)

# BAB I

# PENDAHULUAN

## Latar Belakang

Didalam kebudayaan yang semakin maju diperlukan suatu alat yang mempunyai kemampuan untuk melakukan pekerjaan rutin, menghitung, menyimpan informasi dalam jumlah besar, mengambil data dengan cepat secara acak (random) maupun urut (sequential) dan menyelesaikan persoalan rumit serta banyak perhitungan dalam waktu yang cepat. Dalam hal ini komputer merupakan alat yang memainkan peranan yang sangat besar.

Kehadiran komputer didalam suatu organisasi atau instansi sekolah akan sangat menunjang efisiensi kinerja sehingga akan mendapatkan dampak yang positif didalam suatu aktifitas sekolah maupun instansi lainnya. Komputer adalah sarana yang digunakan untuk membantu mencapai hasil kerja yang maksimal dan dapat menunjang informasi yang cepat dan akurat.

Komputer pada hakekatnya merupakan suatu alat untuk menghitung, sama halnya dengan sempoa, mistar hitung dan sebagainya. Hanya bedanya komputer mempunyai kelebihan dibandingkan dengan alat hitung lainnya berupa kemampuan “mengingat”. Ini berkat adanya bagian dari komputer yang disebut “ingatan” (memory atau strorage), yang akan mengingat atau menyimpan data berupa keterangan – keterangan yang harus dihitung, hasil antara perhitungan atau hasil akhir suatu perhitungan. Disamping itu yang membedakan komputer dengan alat hitung lainnya adalah kemampuannya untuk melakukan pengambilan keputusan yang logis, berkat adanya “ingatan” tadi. Kemampuan-kemampuan inilah yang kemudian dimanfaatkan sebesar-besarnya didalam mengembangkan kemampuan lebih dikenal sebagai alat pengolah data.

Mengandalkan pengolahan data yang baik sangat diperlukan oleh sebuah perusahaan atau organisasi karena dengan pengolahan data yang terkomputerisasi dapat mempercepat pengambilan keputusan oleh pihak diinginkan sehingga organisasi atau pengolahan tersebut dapat meningkatkan kinerjanya.

## Rumusan Masalah

### Bagaimana merancang suatu sistem penerimaan mahasiswa baru berbasis program java?

### Bagaimana agar dapat menciptakan sistem penerimaan mahasiswa baru dengan lebih efisien dan mendapatkan hasil yang maksimal?

## Manfaat Program

### Merancang sistem penerimaan mahasiswa baru dengan program yang berbasis java

### Mempermudah dalam sistem penerimaan mahasiswa baru sehingga hasil yang didapatkan akan lebih efisien dan optimal serta sebagai penerapan perkembangan teknologi.

# BAB II

# LANDASAN TEORI

Dalam program ini digunakan beberapa struktur data dalam java, antara lain:

## Sorting

Sorting adalah proses menyusun elemen – elemen dengan tata urut tertentu dan proses tersebut terimplementasi dalam bermacam aplikasi. Kita ambil contoh pada aplikasi perbankan. Aplikasi tersebut mampu menampilkan daftar account yang aktif. Hampir seluruh pengguna pada sistem akan memilih tampilan daftar berurutan secara ascending demi kenyamanan dalam penelusuran data.

Dalam program ini digunakan salah satu jenis sorting, yaitu bubble sort. Metode bubble sort merupakan metode tersederhana untuk melakukan pengurutan data, tetapi memiliki kinerja yang terburuk untuk data yang besar. Diberi nama “Bubble” karena proses pengurutan secara berangsur-angsur bergerak/berpindah ke posisinya yang tepat, seperti gelembung yang keluar dari sebuah gelas bersoda. Bubble Sort mengurutkan data dengan cara membandingkan elemen sekarang dengan elemen berikutnya. Penukaran baru dilakukan jika suatu kriteria terpenuhi.

## Searching

Pencarian (searching) merupakan proses yang sering digunakan  dalam pengelolaan data. Proses pencarian adalah menemukan nilai (data) tertentu di dalam sekumpulan data yang bertipe sama (baik bertipe dasar atau bertipe bentukan). Search algoritma adalah algoritma yang menerima argument A dan mencoba untuk mencari record yang mana keynya adalah A.

Pada program ini digunakan salah satu jenis searching, yaitu sequential search. Sequential search adalah suatu teknik pencarian data dalam array ( 1 dimensi ) yang akan menelusuri semua elemen-elemen array dari awal sampai akhir, dimana data-data tidak perlu diurutkan terlebih dahulu. Kemungkinan terbaik (best case) adalah jika data yang dicari terletak di indeks array terdepan (elemen array pertama) sehingga waktu yang dibutuhkan untuk pencarian data sangat sebentar (minimal). Kemungkinan terburuk (worst case) adalah jika data yang dicari terletak di indeks array terakhir (elemen array terakhir) sehingga waktu yang dibutuhkan untuk pencarian data sangat lama (maksimal).

Prinsip kerja dari Sequential Searching ini adalah semua data di cek oleh variabel cari. Sequential search dibedakan menjadi dua macam yaitu Pencarian beruntun pada larik tidak terurut dan pencarian beruntun pada larik terurut.

## Linked List

Pengertian Linked list :

* Sekumpulan elemen bertipe sama, yang mempunyai keterurutan tertentu, yang setiap elemennya terdiri dari dua bagian
* Struktur berupa rangkaian elemen saling berkait dimana setiap elemen dihubungkan elemen lain melalui pointer. Pointer adalah alamat elemen. Penggunaan pointer untuk mengacu elemen berakibat elemen-elemen bersebelahan secara logik walau tidak bersebelahan secara fisik di memori.

Link list adalah desain tempat penyimpanan data yang terdiri dari node-node (simpul-simpul) yang saling terhubung

Link list dapat diilustrasikan seperti kereta api, dimana kereta api terdiri dari gerbong-gerbong yang saling terhubung yang dapat mengangkut penumpang. Gerbong disini setara dengan node dalam link list yang berfungsi untuk menyimpan data.

Operasi-operasi yang bisa dilakukan dalam link list yaitu:

### Tambah data (insert)

### Edit data (edit)

### Hapus data (delete)

### Pengurutan data (sorting)

### Pencarian data (searching)

# BAB III

# PEMBAHASAN PROGRAM

## Penjelasan isi dan alur program

Dalam program ini, terdapat beberapa menu awal untuk memudahkan para pengguna atau pendaftar, yaitu antara lain:

### Data pendaftar keseluruhan

### Tambah pendaftar

### Cari data

### Ubah data

### Hapus data

### Urutkan semua pendaftar

### Keluar

Jika kita memilih menu ke-1, maka akan ditampilkan data keseluruhan pendaftar, namun apabila kita belum memasukkan data pendaftar baru maka secara otomatis menu pertama tidak akan menampilkan apapun. Maka dari itu kita harus terlebih dahulu memasukkan data pendaftar baru yang ada pada menu ke-2. Apabila kita memilih menu ke-2, itu adalah menu form pendaftaran, maka akan tampil beberapa tampilan yaitu, Nomor registrasi, Nama, Jenis Kelamin [L/P], Tempat lahir, Tanggal Lahir, Alamat (Kota), dan Asal sekolah. Setiap tampilan tersebut harus kita isi satu persatu agar data yang diminta lengkap. Apabila kita ingin menambahkan data pendaftar baru lebih dari satu, maka kita hanya tinggal memilih menu nomor 2 kembali.

Setelah itu, pada menu ke-3 terdapat menu cari data, dimana menu ini berfungsi sebagai pencari data otomatis. Kita hanya perlu memasukan no registrasi yang sudah dituliskan pada menu nomor 2 yaitu menu form pendaftaran, setelah no registrasi dituliskan, makan akan muncul tampilan data pendaftar yang ingin kita cari. Apabila nomor registrasi yang dimasukkan salah, maka akan ada tampilan “Data tidak ditemukan”, maka kita harus mengulang memasukkan nomor registrasi.

Lalu selanjutnya, apabila kita memilih menu ke-4 maka kita akan mengubah data pendaftar. Disini terdapat pilihan pengubahan berdasarkan nama atau nomor registrasi. Jika kita ingin mengubah nama ataupun nomor registrasi, maka kita akan diminta untuk memasukkan no registrasi pendaftar, apabila kita salah dalam memasukkan nomor registrasi, maka akan ada tampilan “ Data yang dimasukkan salah”, maka kita harus mengulang kembali memasukkan nomor registrasi. Setelah nomor registrasi yang dimasukkan benar, maka akan ada tampilan data seluruh pendaftar yang sudah diubah, baik dari nama ataupun nomor registrasi.

Kemudian, pada menu ke-5 terdapat menu hapus data, menu ini berfungsi apabila kita ingin menghapus salah satu data yang ada. Apabila kita ingin menghapus salah satu data, maka kita akan diminta untuk menginputkan nomor registrasi pendaftar yang ingin dihapus. Setelah memasukkan nomor registrasi yang ingin dihapus, maka secara otomatis sistem akan menghapus data pendaftar yang diinginkan. Apabila kita salah dalam menginputkan nomor registrasi, maka akan ada tampilan “ Data yang anda inputkan salah” maka kita harus mengulang menginputkan nomor registrasi kembali.

Lalu menu ke-6 yaitu menu mengurutkan semua pendaftar. Menu ini hampir sama dengan menu ke-1, namun pada menu ke-6 ini, data yang diurutkan sudah melalui beberapa proses, seperti pengubahan data, penghapusan data. Pada intinya pada menu ke-6 ini adalah tampilan untuk menampilan ataupun mengurutkan keseluruhan data pendaftar baru.

Yang terakhir adalah menu ke-7, yaitu menu keluar. Menu ini digunakan untuk mengakhiri proses apabila kita akan melakukan proses lagi, baik proses penginputan data, pengubaan data dan proses yang lainnya.

## Diagram Alir ( Flow Chart )

## Source Code dan Penjelasan Program

1. Nama kelas untuk menyimpan variable yang akan digunakan dalam program

class PMB

2. Variabel utama yang digunakan dalam program

public int noReg;

public String namaSiswa;

public String jenisKelamin;

public String tempatLahir;

public String tanggalLahir;

public String alamat;

public String asalSekolah;

public PMB next;

3. Objek yang diambil dari variable utama yang akan digunakan dalam fungsi

public PMB (int reg, String nama, String jk, String tmpLahir, String tglLahir, String asal, String aSek) {

noReg = reg;

namaSiswa = nama;

jenisKelamin = jk;

tempatLahir = tmpLahir;

tanggalLahir = tglLahir;

alamat = asal;

asalSekolah = aSek;

next = null;

}

4. Fungsi yang mendeklarasikan variable untuk mengupdate nama dan nomor registrasi

public void setNama (String nama) {

namaSiswa = nama;

}

public void setReg (int reg) {

noReg = reg;

}

}

5. Kelas untuk menambah, melakukan pencarian, update no registrasi dan nama, menghapus, mengurutkan, dan menampilkan data pada program

class OperasiPMB

6. Variabel yang hanya dapat di akses dalam kelas operasiPMB

private PMB head,tail,hasilcari;

1. Variabel kosong yang dapat diisi nilai baru

public OperasiPMB () {

head = null;

}

1. Fungsi tambahData

public void tambahData (int a, String b, String c, String d, String e, String f, String g)

1. Fungsi untuk menambahkan data baru jika data pada variable kosong

{

PMB baru = new PMB (a,b,c,d,e,f,g);

PMB bantu;

if (head==null) {

head = baru;

tail = baru;

}

1. Fungsi untuk menambahkan data baru jika data pada variable telah berisi dan perintah untuk membuat variable kosong yang baru

else {

for (bantu=head; bantu!=null; bantu=bantu.next) {

if (bantu.noReg==a)

break;

}

if (bantu==null) {

tail.next = baru;

tail = baru;

}

else

System.out.println("\n\nNomor Registrasi " + a + " sudah ada");}}

1. Fungsi untuk melakukan pencarian data

public boolean cekData (int cek)

1. Pencarian data berdasarkan kata kunci dengan menelusuri seluruh data, dan apabila kondisi variable tersebut kosong, maka tidak ada data yang dikembalikan

{

PMB bantu;

for (bantu=head; bantu!=null; bantu=bantu.next) {

if (bantu.noReg==cek) {

hasilcari = bantu;

break;

}

}

if (bantu==null)

return false;

else

return true;

}

1. Fungsi menampilkan data hasil pencarian

public void displayCari () {

System.out.print("Nomor Registrasi : " + hasilcari.noReg);

System.out.print("\nNama : " + hasilcari.namaSiswa +" ");

System.out.print("\nJenis Kelamin [L/P] : " + hasilcari.jenisKelamin +" ");

System.out.print("\nTempat Lahir : " + hasilcari.tempatLahir +" ");

System.out.print("\nTanggal Lahir : " + hasilcari.tanggalLahir +" ");

System.out.print("\nAlamat : " + hasilcari.alamat +" ");

System.out.print("\nAsal Sekolah : " + hasilcari.asalSekolah +" ");

}

1. Fungsi untuk mengupdate nama calon mahasiswa baru (pendaftar) dan nomor registrasi

public void ubahNama (int cek, String nama)

public void ubahReg (int cek,int reg)

1. Fungsi untuk melakukan pengecekan data dan apabila kondisi variable tersebut memiliki data, maka data yang lama akan ditumpuk oleh data yang baru

{

PMB bantu;

for (bantu=head; bantu!=null; bantu=bantu.next) {

if (bantu.noReg==cek)

break;

}

if (bantu!=null)

bantu.setNama(nama);

}

{

PMB bantu;

for (bantu=head; bantu!=null; bantu=bantu.next) {

if (bantu.noReg==cek)

break;

}

if (bantu!=null)

bantu.setReg(reg);

}

1. Fungsi untuk menghapus data calon mahasiswa baru

public PMB hapusData (int hapus)

1. Kondisi apabila variable data tidak kosong, jika kata kunci sama dengan data di depan maka data yang sesuai dengan kata kunci akan diubah nilai variabelnya menjadi null atau dikosongkan. Dan terdapat kondisi apabila kata kunci tidak sama dengan data depan, maka penulusuran data pada list tidak sama dengan data depan yang dicari.

{

PMB bantu,bantu2,temp;

if (head!=null) {

if (head.noReg == hapus) {

temp = head;

head = head.next;

temp = null;

return temp;

}

else {

bantu = head;

bantu2 = head;

while (bantu.noReg!=hapus) {

bantu = bantu.next;

while (bantu2.next.noReg!=hapus) {

bantu2 = bantu2.next;}}

1. Kondisi apabila data yang dicari tidak ada yang sama, maka akan dilakukan pencarian dari belakang. Apabila data ditemukan, maka data belakang akan dihapus atau bernilai null. Sedangkan apabila data tidak ditemukan dimanapun, maka nilai yang dikembalikan adalah bernilai null.

if (bantu==tail) {

temp = tail;

tail = bantu2;

tail.next = null;

return temp;

}

else {

temp = bantu;

bantu = bantu.next;

bantu2.next = bantu;

temp = null;

return temp;}}}

else {

System.out.println("Data tidak ditemukan");

return null;

}

}

1. Fungsi untuk melakukan pengurutan (sorting)

public void sortList ()

1. Variabel bantu yang berfungsi untuk penyimpanan sementara dalam melakukan sorting

{

PMB bantu, bantu2;

int tempReg;

1. Proses penyortingan data menggunakan variable pembantu berdasarkan urutan nomor registrasi dari terkecil ke terbesar (ascending)

String tempNama, tempJK, tempTmpLahir, tempTglLahir, tempAsal, tempASek;

for (bantu=head; bantu!=null; bantu=bantu.next) {

for (bantu2=bantu.next; bantu2!=null; bantu2=bantu2.next) {

if (bantu2.noReg<bantu.noReg) {

tempReg = bantu.noReg;

bantu.noReg = bantu2.noReg;

bantu2.noReg = tempReg;

tempNama = bantu.namaSiswa;

bantu.namaSiswa = bantu2.namaSiswa;

bantu2.namaSiswa = tempNama;

tempJK = bantu.jenisKelamin;

bantu.jenisKelamin = bantu2.jenisKelamin;

bantu2.jenisKelamin = tempJK;

tempTmpLahir = bantu.tempatLahir;

bantu.tempatLahir = bantu2.tempatLahir;

bantu2.tempatLahir = tempTmpLahir;

tempTglLahir = bantu.tanggalLahir;

bantu.tanggalLahir = bantu2.tanggalLahir;

bantu2.tanggalLahir = tempTglLahir;

tempAsal = bantu.alamat;

bantu.alamat = bantu2.alamat;

bantu2.alamat = tempAsal;

tempASek = bantu.asalSekolah;

bantu.asalSekolah = bantu2.asalSekolah;

bantu2.asalSekolah = tempASek;}}}}

1. Fungsi untuk menampilkan data keseluruhan

public void displayList () {

if (head != null) {

PMB bantu = head;

int i = 1;

System.out.print("\n============================\n\n");

System.out.println("DATA MAHASISWA BARU");

System.out.print("\n============================\n");

System.out.println("No. Reg\t| Nama\t\t| L/P\t| Tempat,Tgl Lahir\t| Alamat\t| Asal Sekolah");

System.out.println("----------------------------------------------------------------------------------------------");

while (bantu!=null) {

System.out.print(bantu.noReg);

System.out.print("\t| " + bantu.namaSiswa);

System.out.print("\t| " + bantu.jenisKelamin);

System.out.print("\t| " + bantu.tempatLahir + ", " + bantu.tanggalLahir);

System.out.print("\t| " + bantu.alamat);

System.out.println("\t| " + bantu.asalSekolah);

bantu = bantu.next;

i++;

}

System.out.println("");

}

else

System.out.println("List Kosong.");

}

}

1. Library **Scanner** untuk menginputkan data

import java.util.Scanner;

1. Kelas utama yang mengandung program utama

class PMBApp

1. Pendeklarasian metode scanner untuk membuat variable baru

Scanner scn = new Scanner (System.in);

Scanner scr = new Scanner (System.in);

1. Pembuatan list

OperasiPMB list = new OperasiPMB ();

1. Data awal yang sudah disimpan dalam program

int menu;

do {

System.out.println("\n\n\n\n--------------------------------");

System.out.println("PENERIMAAN MAHASISWA BARU \nPROGRAM STUDI TEKNIK INFORMATIKA \nFAKULTAS SAINS DAN TEKNOLOGI \nUIN SGD BANDUNG");

System.out.println("================================\n");

System.out.println("[1] DATA PENDAFTARAN KESELURUHAN");

System.out.println("[2] TAMBAH PENDAFTAR");

System.out.println("[3] CARI DATA");

System.out.println("[4] UBAH DATA");

System.out.println("[5] HAPUS DATA");

System.out.println("[6] URUTKAN SEMUA PENDAFTAR");

System.out.println("[7] KELUAR");

System.out.println("--------------------------------\n");

System.out.print("Pilih Menu : ");

menu = scn.nextInt ();

switch (menu)

1. Menu pertama : Data Pendaftaran Keseluruhan

case 1 : {

list.displayList ();

}

break;

1. Menu kedua : Tambah pendaftar, yaitu untuk menambahkan data ke dalam list

case 2: {

System.out.println("--------------------------------");

System.out.println("FORM REGISTRASI MAHASISWA BARU");

System.out.println("--------------------------------\n");

System.out.print("Nomor Registrasi : ");

int reg = scn.nextInt ();

System.out.print("Nama : ");

String nama = scr.nextLine ();

System.out.print("Jenis Kelamin [L/P] : ");

String jk = scr.nextLine ();

System.out.print("Tempat Lahir : ");

String tmpLahir = scr.nextLine ();

System.out.print("Tanggal Lahir [dd/mm/yyyy]: ");

String tglLahir = scr.nextLine ();

System.out.print("Alamat (Kota) : ");

String asal = scr.nextLine ();

System.out.print("Asal Sekolah : ");

String aSek = scr.nextLine ();

list.tambahData (reg, nama, jk, tmpLahir, tglLahir, asal, aSek);}

break;

1. Menu ketiga : Cari Data Mahasiswa

case 3: {

System.out.println("--------------------------------");

System.out.println("CARI DATA MAHASISWA BARU");

System.out.println("--------------------------------\n");

System.out.print("Masukkan Reg : ");

int reg = scn.nextInt ();

if (list.cekData (reg)) {

System.out.println("Data Ditemukan");

list.displayCari ();

}

else

System.out.println("Data tidak ditemukan");

}

break;

1. Menu Keempat : Ubah data

case 4: {

System.out.println("--------------------------------");

System.out.println("UBAH DATA");

System.out.println("--------------------------------\n");

System.out.println("Ubah berdasarkan : ");

System.out.println("[1] Nama");

System.out.println("[2] Nomor Registrasi");

System.out.print("Pilih : ");

int pil = scn.nextInt ();

if (pil==1) {

System.out.println("========= UBAH NAMA ========\n");

System.out.print("No Reg : ");

int REG = scn.nextInt ();

if (list.cekData (REG)) {

System.out.print("Ubah Nama : ");

String nama = scr.nextLine ();

list.ubahNama (REG,nama);

list.displayList ();}

else

System.out.println("Data yang dimasukkan salah");}

else if (pil==2) {

System.out.println("========= UBAH NO REG ========\n");

System.out.print("No Reg : ");

int REG = scn.nextInt ();

if (list.cekData (REG)) {

System.out.print("Ubah No Registrasi : ");

int noReg = scn.nextInt ();

list.ubahReg (REG,noReg);

list.displayList ();}

else

System.out.println("Data yang dimasukkan salah");}

else

System.out.println("Data yang anda inputkan salah :");}

break;

1. Menu Kelima : Hapus Data Mahasiswa

case 5: {

System.out.println("--------------------------------");

System.out.println("HAPUS DATA MAHASISWA BARU");

System.out.println("--------------------------------\n");

System.out.print("Hapus Nomor Registrasi : ");

int reg = scn.nextInt ();

if (list.cekData (reg)) {

list.hapusData (reg);

System.out.println("Data Terhapus");

list.displayList ();}

else

System.out.println("Data yang anda inputkan salah :");}

break;

1. Menu Keenam : Mengurutkan Data Seluruh Pendaftar

case 6: {

list.sortList ();

list.displayList ();}}

}

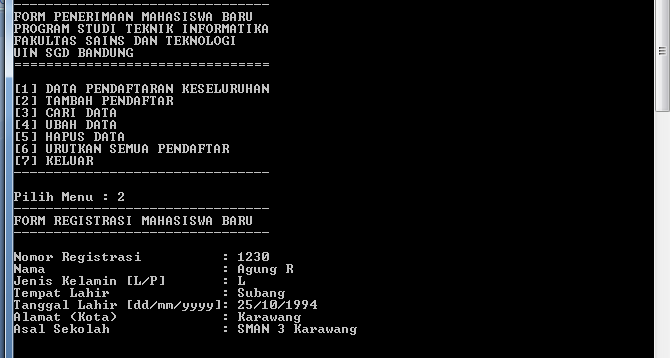
}

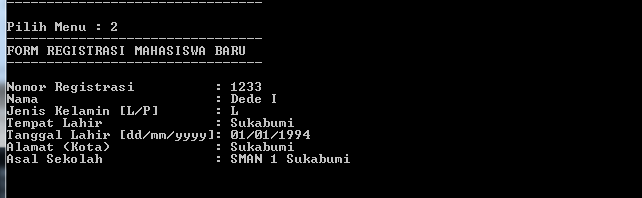
1. Menu Ketujuh : Keluar dari program

while (menu!=7);}}}}

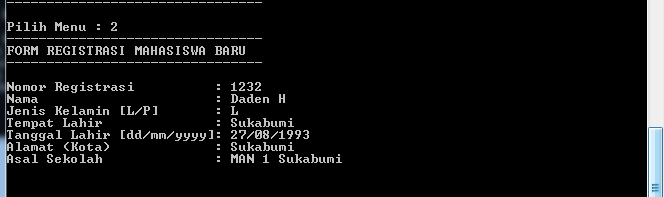
## Tampilan Hasil Eksekusi

### Tambah Pendaftar

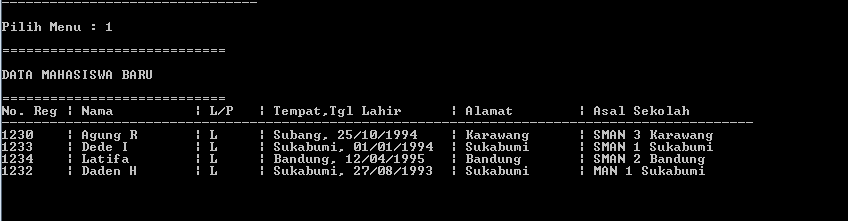








### Data Pendaftar Keseluruhan ( Data Mahasiswa Baru )



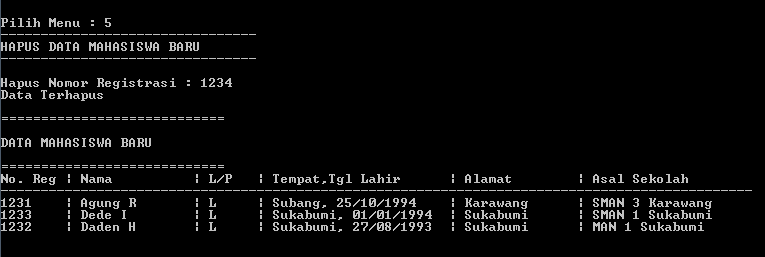
### Cari Data Mahasiswa



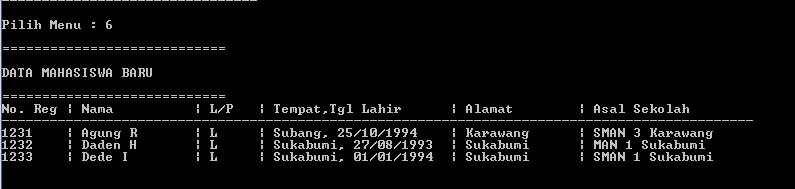
### D:\PIC\7.pngUbah Data Mahasiswa

1. Urutkan Semua Pendaftar

### Hapus Data Mahasiswa



### Urutkan Data Mahasiswa



# BAB IV

# KESIMPULAN DAN SARAN

## Kesimpulan

Mengandalkan pengolahan data yang baik sangat diperlukan oleh sebuah perusahaan atau organisasi karena dengan pengolahan data yang terkomputerisasi dapat mempercepat pengambilan keputusan oleh pihak diinginkan sehingga organisasi atau pengolahan tersebut dapat meningkatkan kinerjanya.

Dalam program ini digunakan beberapa struktur data dalam java, antara lain searching, sorting, serta linked list.

Dalam program ini, terdapat beberapa menu awal untuk memudahkan para pengguna atau pendaftar, yaitu antara lain: Data pendaftar keseluruhan, tambah pendaftar, cari data, ubah data, hapus data, urutkan semua pendaftar dan menu keluar.

Program ini dibuat untuk memudahkan dalam sistem pendaftaran mahasiswa baru agar didapatkan hasil yang optimal serta efisiensi waktu.

## Saran

### Kepada para pembaca agar dapat lebih mengembangkan program java terutama dalam bidang administrasi, karena sistem administrasi termasuk salah satu bidang yang sangat membutuhkan program komputerisasi dalam pengerjaannya.

### Para pembaca khususnya programmer dapat menggunakan algoritma sebagai solving dalam masalah logika dan masalah matematika.

# DAFTAR PUSTAKA

### <http://deva-radhi.blogspot.com/>

### <http://tasleptop.wordpress.com/2012/03/07/pencarian-searching-sequential-search/>

### <http://wwwpemograman-hotang.blogspot.com/2011/05/makalah-struktur-data.html>

### <http://fennyjh3.blogspot.com/2013/05/linked-list-java.html>